

## Relación de Ejercicios en Android (1)

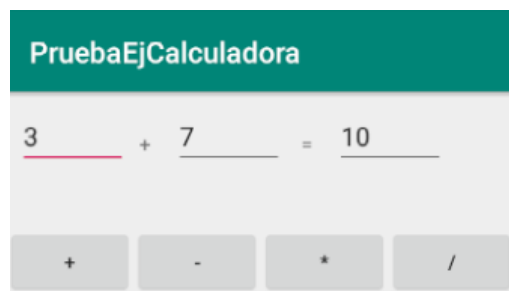
---

- 1) Escribe el programa **ConversorCentímetrosPulgadas**, que nos permitirá transformar valores de una unidad a la otra. La interfaz constará de un par de *EditText* para poner los centímetros y las pulgadas y un par de botones para cada sentido de la conversión.



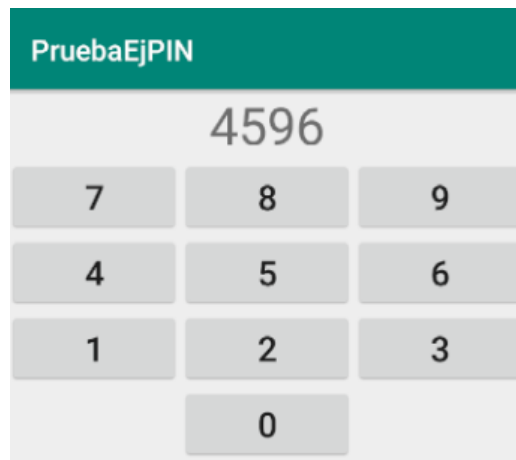
Cuando pulsemos el botón de “centímetros a pulgadas” se cogerá el valor que haya escrito en centímetros y nos mostrará las pulgadas correspondientes y viceversa. Cuidado de que el programa no explote, que queda muy feo en Android.

- 2) Escribe el programa **Calculadora**. La interfaz de este programa contará con dos *EditText* donde escribiremos los operandos y un tercero en el que pondremos el resultado. Entre ambos, pondremos un par de *TextView* con los operadores. Además, tendremos cuatro botones correspondientes cada uno a una operación aritmética: suma, resta, multiplicación, división. Al pulsar cualquiera de los botones, se realizará la operación entre los valores de los dos primeros *EditText*, pondremos el resultado en el tercer *EditText* y cambiaremos el operador que hay entre ambos para reflejar la operación que se ha realizado.



Importante: como ya estamos haciendo aplicaciones con interfaz gráfico, hay que tener mucho cuidado de que nuestro programa no explote durante la ejecución. Para ello, se recomienda comprobar que los *EditText* contienen valores correctos antes de intentar hacer la operación.

- 3) Escribe el programa **Teclado PIN**. En este programa, tendremos un teclado formado por botones correspondientes a los 10 dígitos y podremos pulsar en ellos para ir introduciendo un código, que se mostrará en un *TextView*. En el ejemplo he usado un *TableLayout*, que facilita el colocar las cosas en cuadrículas.



Para ello, sería recomendable escribir solamente una función *onClick* y asignar esa misma función a todos los botones, ya que hacen todos más o menos lo mismo. A *onClick* se le pasa un parámetro *View v*, que corresponde al elemento que ha lanzado el *onClick* (en este caso, un botón). Simplemente convirtiendo ese *View* en un *Button*, podremos acceder a todas sus propiedades.

```
Button b = (Button)v;
```

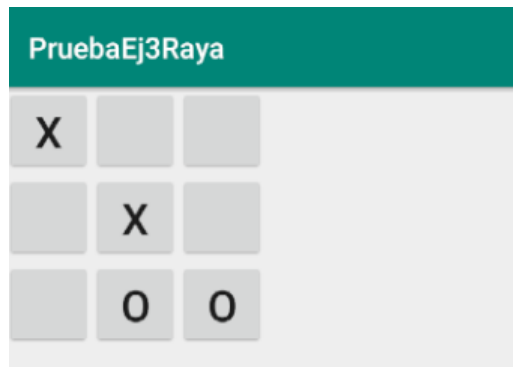
Para que sea más interesante, el usuario deberá escribir un código de 4 dígitos (por ejemplo, 1-2-3-4). Si escribe el código correcto, cambiaremos el color del *TextView* a verde, mientras que, si se escribe uno incorrecto, se borrarán los 4 números para poder escribir de nuevo otro código.

- 4) Escribe el programa **Adivino** que tratará de adivinar el número entre 1 y 100 que piense el usuario. Tras haber pensado el número, el usuario pulsará el botón *Empezar*. A continuación, el programa mostrará un número entre 1 y 100 (se recomienda empezar por el 50). El usuario deberá pulsar el botón *Menor* si el número que él ha pensado es menor que el que muestra el programa, el botón *Mayor* si el número es mayor y el botón *Igual* si es el número que ha pensado.



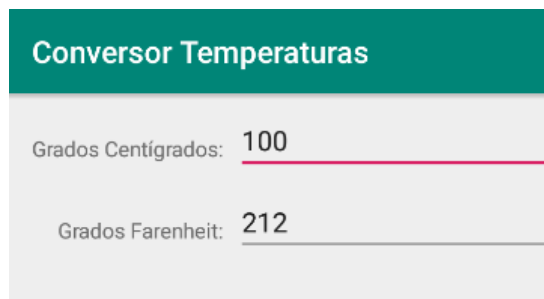
De inicio, el botón *Empezar* será el único que se puede pulsar. Una vez empieza el juego, se activarán el resto de los botones. Cuando se haya acertado el número, se deshabilitarán los tres botones y se volverá a activar el botón *Empezar* por si se quiere empezar de nuevo la partida.

- 5) Escribe el programa **TresEnRaya** que nos permitirá jugar a las tres en raya contra el ordenador usando una interfaz visual.



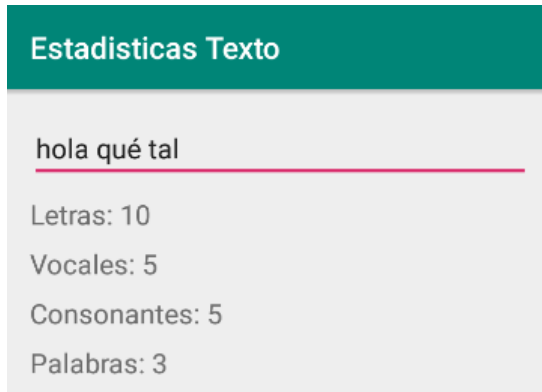
Para simplificar, sólo tendrá un modo: usuario vs. ordenador y siempre empezará el usuario. Cuando el usuario pulse en un botón, se marcará con el símbolo X (siempre y cuando no esté ya ocupado) y entonces moverá el ordenador con el símbolo O. El juego terminará cuando uno de los dos gane la partida o haya tablas. Podéis mostrar el resultado de la partida usando un *TextView*.

- 6) Escribe el programa **Conversor Temperaturas**, que nos convertirá entre grados centígrados y grados fahrenheit. La interfaz contará con dos *EditText* de tipo numérico con signo (para que se puedan poner negativos) y un par de *TextView*.



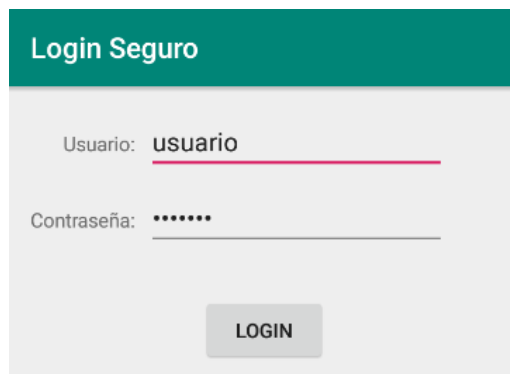
A diferencia del conversor que hicimos anteriormente, aquí no pulsaremos ningún botón para que se produzca la conversión, sino que cada vez que escribamos algo en cualquier de los dos *EditText* se actualizará el otro *EditText* con el valor correspondiente.

- 7) Escribe el programa **Estadísticas Texto**, que nos calculará unas estadísticas del texto que le escribamos en un *EditText*. Además de ese *EditText*, tendremos varios *TextView* donde mostraremos las estadísticas.

The screenshot shows a mobile application interface with a teal header bar containing the text "Estadísticas Texto". Below the header is a light gray area. At the top of this area is an *EditText* field containing the text "hola qué tal". Below the text field are four *TextView* elements displaying statistics: "Letras: 10", "Vocales: 5", "Consonantes: 5", and "Palabras: 3".

El programa nos calculará el número de letras, de vocales, de consonantes y de palabras del texto que escribamos en el *EditText*. Estas estadísticas se recalcularán cada vez que cambie el texto.

- 8) Escribe el programa **Login Seguro**. Este programa constará de varias *Activities*. En la *MainActivity*, pondremos un par de *EditText* donde preguntaremos un nombre de usuario y una contraseña. Si los datos son correctos, pasará a otra *Activity*.

The screenshot shows a mobile application interface with a teal header bar containing the text "Login Seguro". Below the header is a light gray area. It contains two *EditText* fields: the first is labeled "Usuario:" and contains the text "usuario"; the second is labeled "Contraseña:" and contains seven dots. Below these fields is a gray button with the text "LOGIN".

Tendremos dos usuarios: el usuario normal (usuario/usuario) y el administrador (admin/admin). Cuando acceda el usuario normal, iremos a otra *Activity* con el fondo verde que ponga: "Bienvenido, usuario". Cuando acceda el administrador iremos a otra *Activity* distinta con el fondo rojo que ponga "ADMIN". Si el usuario y contraseña no son válidos, pondremos un *Toast* dando un mensaje de alerta.

- 9) Escribe el programa **Tiradas de dados**. Este programa nos permitirá simular tirada de varios dados de diferentes caras y nos dará el resultado de las tiradas y la suma total.

The screenshot shows a mobile app titled "Tiradas de dados". It has two columns for input: "Número de dados" and "Número de caras". Under "Número de dados", there are three input fields with values 7, 8, and 9. Under "Número de caras", there are two input fields with values 12 and 20. Below these inputs is a button labeled "TIRAR". At the bottom, the app displays the results: "Tiradas: 8, 8, 11, 16, 2, 9, 2, 1" and "Suma: 57".

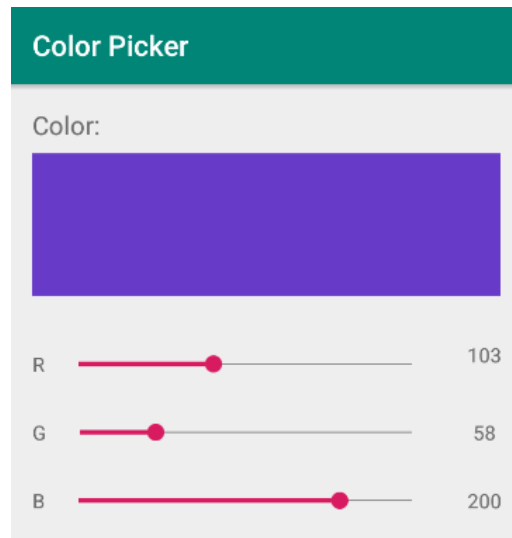
Para seleccionar el número de caras de los dados usaremos dos *NumberPicker*, mientras que el resultado que se muestra al pulsar el *Button* lo mostraremos en sendos *TextView*. El número de dados irá de 1 a 10 mientras que el número de caras lo elegiremos entre los valores: 4, 6, 8, 10, 12 y 20.

- 10) Escribe el programa **Conversor de monedas**. El programa nos permitirá elegir de qué moneda a qué moneda queremos convertir por medio de *RadioButton* e introduciremos la cantidad que queremos convertir en un *EditText*.

The screenshot shows a mobile app titled "Conversor de monedas". It has a section "Convertir de:" with three radio buttons: "Euro €" (selected), "Dolar \$", and "Yen ¥". Below this is a section "a:" with three radio buttons: "Euro €", "Dolar \$", and "Yen ¥" (selected). At the bottom, there is an input field with the value "500" and a label "€", followed by an equals sign and the result "61520.0 ¥".

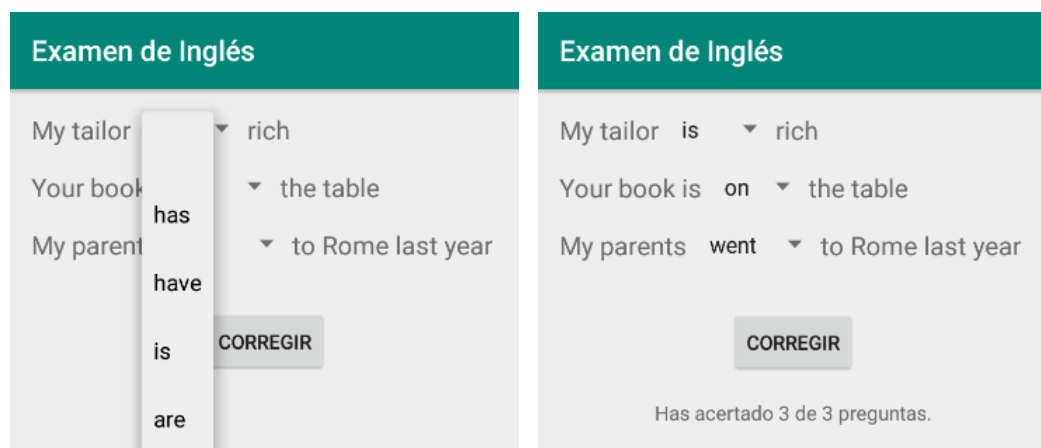
El cálculo se actualizará cada vez que hagamos click en un *RadioButton* o cada vez que se cambie el texto del *EditText*. Como siempre, cuidado con que no explote el programa.

- 11) Escribe el programa **Color Picker**. Este programa nos permitirá seleccionar un color seleccionando los tres valores numéricos RGB y nos mostrará ese color por pantalla. Para seleccionar los tres valores usaremos sendas *SeekBar*.

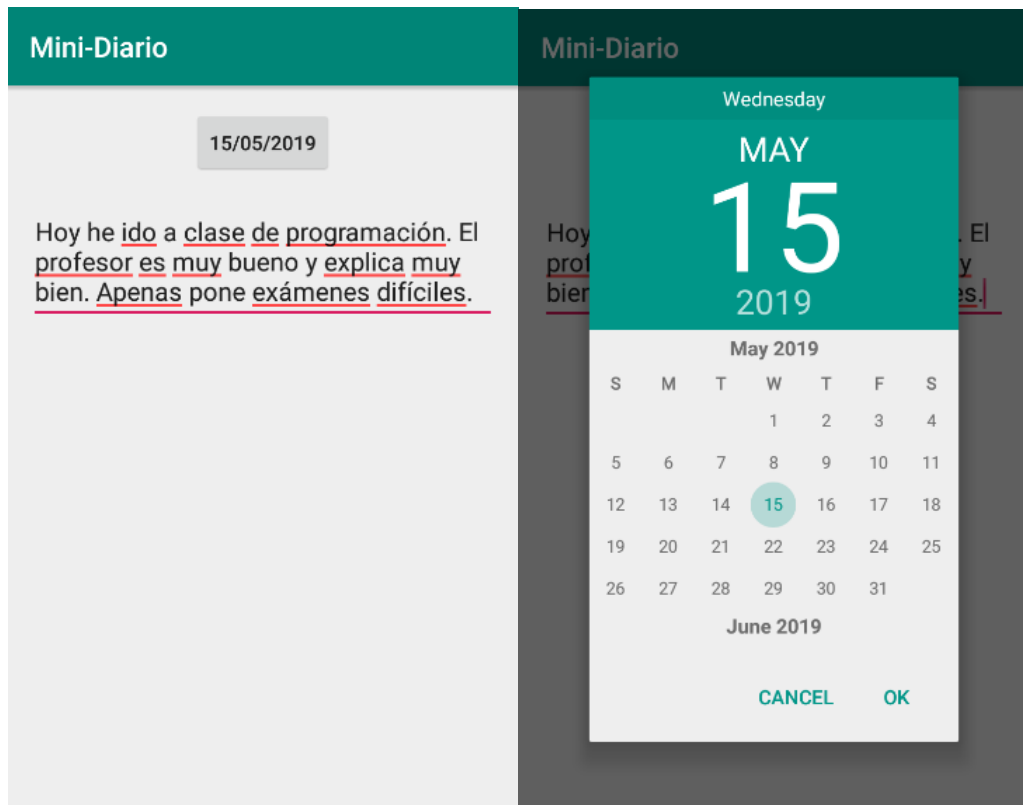


Al mover las *SeekBar* se actualizarán los valores numéricos que hay a la derecha de cada una en un *TextView* y también cambiará el color que se muestra (yo lo he puesto en otro *TextView* usando el método *setBackgroundColor*).

- 12) Escribe el programa **Examen de Inglés**. En este programa, presentaremos al usuario un examen de completar las frases con varias opciones. Para mostrar las opciones usaremos el elemento *Spinner*. El usuario completará las opciones y pulsará el *Button* de corregir tras lo cual se mostrará el resultado.



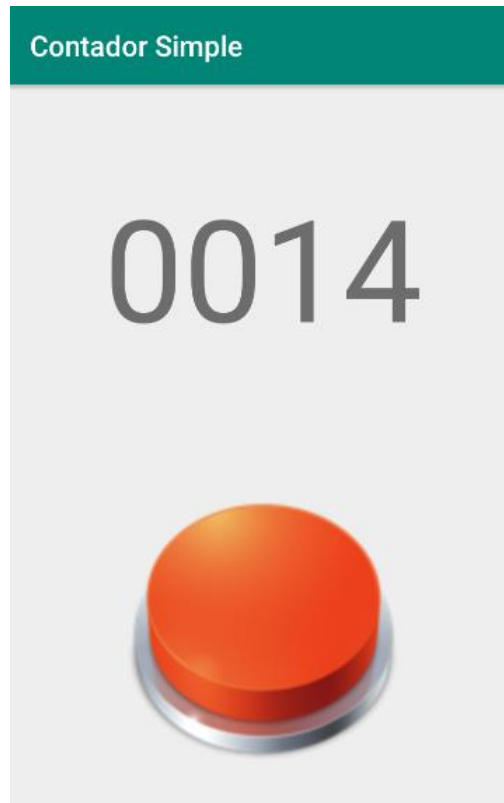
- 13) Escribe el programa **Mini-Diario**. Este programa nos permitirá llevar un diario de todo lo que pase en nuestra vida. Tendrá un *Button* con el que podremos elegir una fecha y un *EditText* para escribir lo que queramos en cada día.



Al iniciar el programa, el *Button* nos mostrará la fecha de hoy y en el *EditText* aparecerá el texto correspondiente al día de hoy. Pulsando el *Button* nos saldrá un *DatePickerDialog* que nos permitirá elegir una nueva fecha. Al pulsar la fecha, se cargará el texto correspondiente al nuevo día. Además, tendremos que guardar el texto del día actual (preferentemente, sólo se guardará si ha cambiado).

El texto de cada día se guardará en archivos (de texto o binarios, me da igual). Cada día se guardará en un archivo con el nombre “dd-mm-yyyy” (para el ejemplo: “15-05-2019”).

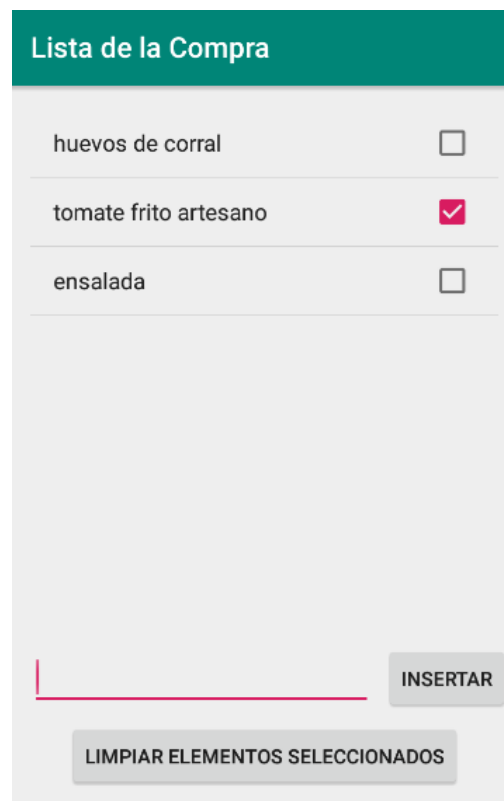
- 14) Escribe el programa **Contador Simple**. Simplemente sumará uno a un contador cada vez que pulsemos un botón y lo mostrará en pantalla.



El botón del contador lo haremos con un *ImageButton* para poder poner un icono chulo, mientras que el texto lo mostraremos en un *TextView*. Para reiniciar el contador a cero y volver a empezar, haremos un click largo sobre nuestro botón (el evento lo pondremos en el *setOnLongClickListener*). Además, el programa deberá guardar el valor actual del contador, de forma que si salimos del programa y volvemos a entrar deberá continuar por donde lo dejamos (esto lo haremos con *SharedPreferences*).



- 15) Escribe el programa **Lista de la Compra**. En este programa tendremos un *ListView* con *CheckBoxes* en el que iremos insertando los valores de la lista de la compra. También tendremos un *EditText* en el que pondremos los nuevos valores y un *Button* para insertarlos.



Para complicarlo un poco, pondremos un segundo *Button* con el que borraremos los elementos seleccionados de la lista y dejaremos sólo los que no están seleccionados.

También tendremos que guardar la lista cuando salgamos del programa y volverla a cargar al iniciar. Como tenemos que guardar tanto los valores de texto como los valores booleanos de los *CheckBoxes* yo recomiendo usar un fichero binario con *DataInputStream* y *DataOutputStream*, pero podéis usar uno de texto si queréis.

Para leer al iniciar el programa, lo ponéis en el *onCreate* de nuestra *Activity* y para guardar al salir lo más apropiado es en el *onPause* (que tenéis que añadir vosotros).